CONSTRAINTS IN AUTHORING BIM COMPONENTS: RESULTS OF LONGITUDINAL INTEROPERABILITY TESTS

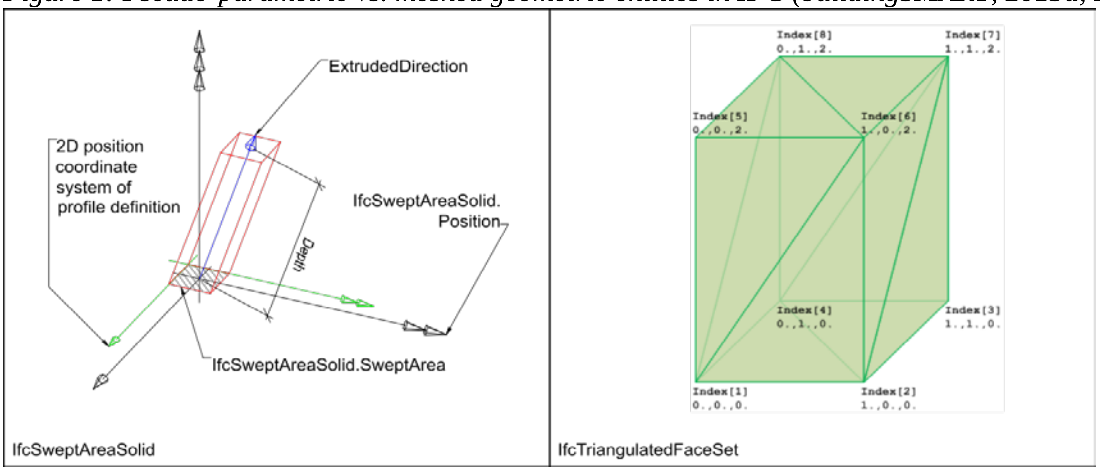
INTRODUCTION

As Building Information Modelling (BIM) becomes more prevalent in the construction industries of the world some of the practical problems of authoring and sharing models are also becoming more evident. It is generally accepted that BIM-related technologies offer considerable advantages to many, perhaps most, participants in the construction sector (Eastman et al., 2011). In the UK, for example, the Government has mandated the use on all its projects of "fully collaborative 3D BIM" (Cabinet Office, 2011) and supported the development of standards to support the industrial uptake (BIM Task Group et al., 2016).
A fundamental limiting factor in the uptake of BIM is the issue of interoperability, defined by the International Alliance for Interoperability (IAI, now 'buildingSMART') as 'an environment in which computer programs can share and exchange data automatically, regardless of the type of software or where the data may be residing' (Fischer and Kam, 2002, p. 14). Currently, there exists a whole range of commercially available BIM software platforms that have specialised to suit the functional needs of their main users (architects, structural engineers, services engineers, constructors, and so on) and consequently differ structurally and semantically. The aspirational ideal of fully collaborative BIM presumes a single model, allowing the full integration of all aspects of the design and further, for the same information to be re-used in the delivery and operation of the constructed facility (UK Treasury and Cabinet Office, 2016. p. 7). To do this effectively, secure and reliable exchange of data is essential. It is this requirement that underlies the concept of 'interoperability' (Yang and Zhang, 2006). As Cerovsek points out a key issue has been how to achieve 'inter-operability between multiple models and multiple tools that are used in the whole product lifecycle' (Cerovsek, 2011, p. 224) and BIM usage is still largely restricted to coordinated models that relate to the contribution of each of the disciplines involved. Currently, as evidenced by NBS's National BIM Report (NBS, 2017) full collaboration is still not a reality. For some time the recognised basis of BIM interoperability has been the system of Industry Foundation Classes (IFCs) designed by the International Alliance for Interoperability and maintained by buildingSMART (Tolman, 1999; Fischer and Kam, 2002). However, the mere presence of IFC is not sufficient for overcoming the problems of interoperability, and, for some critics, data exchange remains 'unreliable and unpredictable' (e.g. Sacks et al., 2010, p. 420). Fischer and Kam (2002, p. 40) identify such problems, particularly when they result in 'geometric misrepresentations across different software packages reading the same IFC source file'.

The problem of standard component libraries

The problem is perhaps most acute when it comes to the effective authoring and use of standard building component libraries for BIM; a development that has, for some time been seen as to have significant potential for improving the productivity of designers and specifiers (Howard and Björk, 2008). Demand for library components from the industry and construction product manufacturers is high, with 93% of practitioners stating they "need access to well-structured generic digital objects" (NBS, 2017, p. 20) and there are organisations currently attempting to author library components that can be delivered across multiple BIM platforms with the minimum amount of re-authoring. Buildings contain numerous components – windows, doors, etc. - that are standardly procured from their manufacturers. However, when they exist within a proprietary BIM tool, these components are native to that particular tool and not easily shared between different BIM platforms; for example, a boiler component authored in one tool is unlikely to be useable in another BIM tool. This leads to a major weakness in data sharing and restricts any efficiency gains that the industry may make through data reuse, both in terms of content creation and subsequent delivery and management. Worse, when the transfer of data appears to work, but is incorrect, the reliability of the whole process is undermined. This has arguably contributed to the relatively slow uptake of BIM standards to date and hampered the efforts of buildingSMART to support the development of library standards through IFC. The current releases of the buildingSMART data model (Ifc2x3 and Ifc4) do not support explicit parametric objects and therefore do not allow the exchange of object's geometric constraints. Although some of these features are under development in the upcoming Ifc5 standard (buildingSMART, 2017), it is likely that for the next few years the industry will need to create library objects within the constraints of the current buildingSMART data standards. In these, whilst explicit parametric behaviour is not supported, implicit parametric behaviour can be approximated by careful selection of the available geometric objects. For example, the IFC-Extruded-Area-Solid defines a solid in terms of a footprint, an extrusion direction and a depth of extrusions, as illustrated in Figure 1.

Figure 1: Pseudo-parametric vs. meshed geometric entities in IFC (buildingSMART, 2013a, 2013b)



Essentially the profile, depth and direction allow the object to behave in a simple parametric way by determining the shape of the resulting object on the single values, as opposed to the use of geometries such as IfcTriangulatedFaceSet, which would require the changing of several individual vertex points to determine its shape. As reported by Fischer and Kam (2002, p. 25) considerable efficiency gains can be made by authoring library components through these implicitly parametric geometric objects.

RESEARCH AIMS AND METHOD

This study investigated the issues around authoring standard reusable BIM components that can be delivered across multiple platforms using IFC as a single source of content. The primary aim was to determine what constraints exist for modelling library components to support current BIM platforms, and to illustrate their effects.

Selecting the software applications

In 2012 we reviewed the BIM software applications that claimed support for the Ifc2x3 standard and these claimants were then filtered to include only those that had some form of geometric presentation and/or authoring engine. These are identified in Table 1. Whilst other platforms exist, it is suggested that these are representative and typical of those currently available for use in the Construction Industry. Alongside the commercially-available BIM software applications, listed in Table 1, the open-source software IFC viewer XbimXplorer (xBimTeam, 2017), was adopted as a reference test, as the authors had previously found it to have performed well in trial tests of this nature and its open source nature provided transparency into its geometric modelling operations. A reference test suite has been published along with the source of a geometry engine that correctly processes these geometries and a viewer to examine the results. This is available at http://docs.xbim.net/research/ifc2x3-test-harness.html.

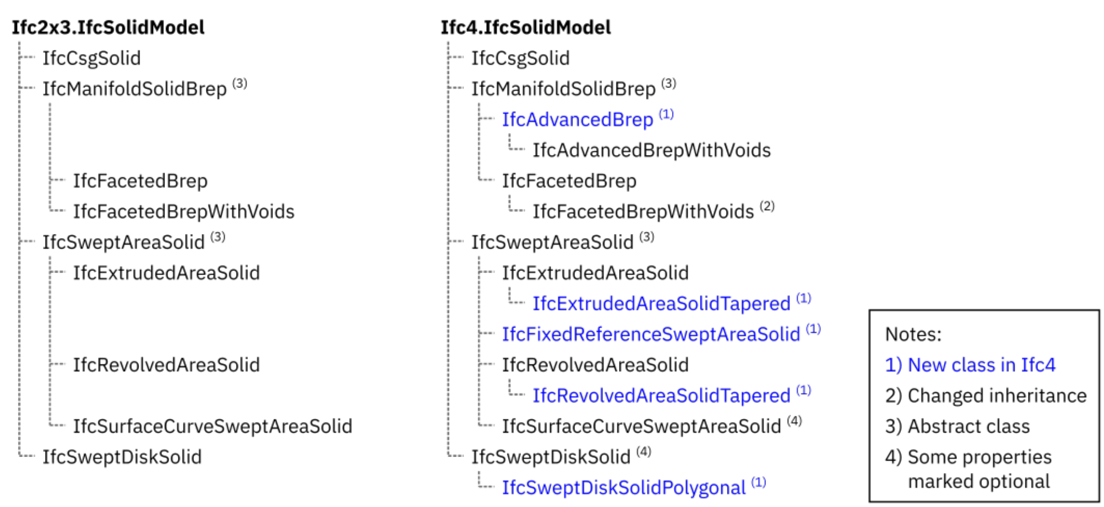
Table 1: Selection of BIM Platforms

Author	BIM Tool	2012 test version	2017 test version	Editing
Bentley	AECOsim	AECOsim v8i	View v. 8i	x
Graphisoft	ArchiCAD	ArchiCAD 16	ArchiCAD 20	x
Nemetcheck	DDS Cad Viewer	V 8.0	V 13.0	
Karlsruhe Institute of Technology	FZKViewer	V 2.2	V 4.8	
	GTDS Viewer	v 0.6	replaced by FZKViewer	
	IfcStorey Viewer	v2.2b		
Autodesk	Revit	v2012	v2017	x
Nemetcheck	Solibri Model Viewer	v7.1	v9.7	
Trimble	Tekla Structures	v18.1	v2017sp2	x
	Tekla BIMsight	v2012	v1.99	
Nemetcheck	VectorWorks	v2012	v2017	x

Selection of relevant geometry cases

The focus of this study is the authoring of building components. These can be defined as discrete compound geometric solid objects that are inserted from a library into a construction project model. Typically these are items which are manufactured off-site and installed during construction; examples would be a boiler, light fitting, door or window. Their three-dimensional visual representation would be described using a combination of the classes represented in Figure 2, which also shows the changes occurred between the schemas of Ifc2x3 and Ifc4.

Figure 2: IFC Solid Model Representations



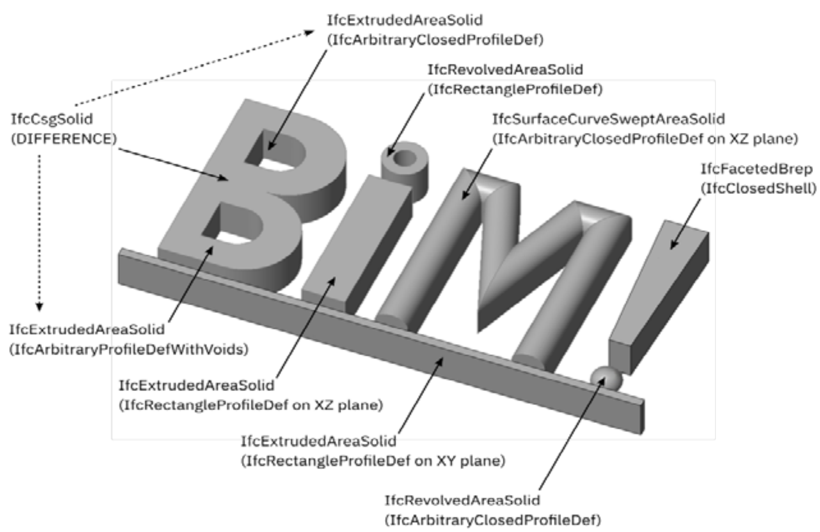
Since the test model was developed ahead of the Ifc4 release, only solid model representation entities defined in buildingSMART data standard Ifc2x3 were included in the tests. In order to develop a representative test we investigated over 200 IFC files produced in professional BIM workflows; this resulted in a clear prevalence of geometries defined with a simplified form of Boundary Representation

(*Brep*) where all faces are planar and every face bound is defined as a polyline. This type of solid is defined as *IfcFacetedBrep* or *IfcFacetedBrepWithVoids*, (which rely on the same underlying geometric functions). Since not a single instance was found of a geometry using the format of *IfcSweptDiskSolid* in the sample, this representation was excluded from the tests as it was prima facie unlikely to give any interoperability concerns in actual BIM workflows. Therefore, five solid geometry representations (*IfcCsgSolid*, *IfcExtrudedAreaSolid*, *IfcRevolvedAreaSolid*, *IfcFacetedBrep* and *IfcSurfaceCurveSweptAreaSolid*) were used to create a simple test reference model that was designed to validate the correct interpretation of geometry and topology. At the time of authoring all of their respective topological shapes were supported as output format by the majority of the available BIM tools.

Design of the Geometric Test Model

A single compound geometric element was constructed that embodied each of these geometries in a form that assisted visual assessment (as well as providing a pleasing irony). This is illustrated in Figure 3 with annotation to indicate the underlying IFC geometries being tested.

Figure 3: Geometry Test Model



The objective was to define individual parts of the model that could provide evidence for correct support for the following six IFC solid cases:

- *IfcCsgSolid* (the letter "B")
- *IfcExtrudedAreaSolid* (the body of the letter "i")
- *IfcRevolvedAreaSolid* (the dot of the letter "i")
- *IfcSurfaceCurveSweptAreaSolid* (the letter "M")
- *IfcFacetedBrep* (the body of the exclamation mark)
- *IfcRevolvedAreaSolid* (the dot of the exclamation mark)

The difference between the two *IfcRevolvedAreaSolid* instances stands in central void of the dot of the letter "i" due to the offset position of the revolution axis. The underlining block upon which the model sits was included only to assist visual comparison, and was developed with the typical geometries of a standard wall (an *IfcExtrudedAreaSolid* of an *IfcRectangleProfileDef* lying on the XY plane) to maximize the likelihood of correct interpretation; its rendering was excluded from the analysis of results. The construction of each of the solid geometries has been designed to validate the following measures of fidelity:

- Visual Presence
- Geometric Equivalence (Cartesian composition)
- Topological Equivalence (Relational composition)

In this test case, measurement of Semantic Fidelity (Element, Material, Properties etc.) and Presentation (Texture, Colour, Line Style etc.) have been excluded (to be the subject of further separate investigation).

RESULTS

The tests were performed by observing the results of importing the IFC files containing all of the composite test model's geometric entities into each of the listed BIM platforms. A 'visual presence' test was performed on the results and a pass was given where any geometry was visible and appeared to be the same as that of the test case. The following Table 2 summarizes the outcomes of each platform's ability to interpret the IFC geometries correctly. Screenshots of resultant representations of the test model are presented (as Figures 4-23) in an appendix to the text of the paper with Figure 24 showing the correct representation from XbimXplorer, which was used to create the reference test suite (available at http://docs.xbim.net/research/ifc2x3-test-harness.html) and accordingly passed all tests. The supplier names of the products listed in Table 2 have been omitted, but can be seen by referring to the earlier, Table 1.

Table 2: Pass/Fail Summary for all geometries of the test model (x signifies a pass)

Product	B	i body	i dot	M	! body	! dot	Pass count and (%)	Entire BiM!	Screen shot
AECOsim v8i		x			x		2 (33%)		Fig. 4
View v. 8i 2017		x	x	x	x	x	5 (83%)		Fig. 5
ArchiCAD 16		x	x				2 (33%)		Fig. 6
ArchiCAD 20		x	x	x	x		4 (67%)		Fig. 7
Cad Viewer 2012		x			x		2 (33%)		Fig. 8
Cad Viewer 2017		x			x	x	3 (50%)		Fig. 9
FZKViewer v2.2		x	x		x	x	4 (67%)		Fig. 10
FZKViewer v4.8	x	x	x	x	x	x	7 (100%)	x	Fig. 11
GTDSViewer 0.6	x	x					2 (33%)		Fig. 12
	Discontinued software has not been re-tested in 2017								
IfcStorey Viewer[4]		x					1 (17%)		Fig. 13
	Discontinued software has not been re-tested in 2017								
Revit 2012		x			x		2 (33%)		Fig. 14
Revit 2017		x			x		2 (33%)		Fig. 15
Solibri SMV 7.1		x	x				2 (33%)		Fig. 16
Solibri SMV 9.7	x	x	x	x	x	x	7 (100%)	x	Fig. 17
Tekla Structures 18.1	x	x	x				3 (50%)		Fig. 18
Tekla Structures 2017	x	x	x		x	x	5 (83%)		Fig. 19

Tekla BIMsight 2012	x	x	x				3 (50%)		Fig. 20
Tekla BIMsight 2017	x	x	x		x	x	5 (83%)		Fig. 21
VectorWorks 2012		x					1 (17%)		Fig. 22
VectorWorks 2018	x	x					2 (33%)		Fig. 23

As Table 12 shows, in the 2012 tests all the tools accurately displayed the body of the “i” in BIM (*IfcExtrudedAreaSolid*). This was not initially the case with ArchiCAD 16, but on further examination it was observed that there had been a total failure in reading the IFC file as a compound entity, resulting in all six geometries being rejected as a single entity due to an internal error in processing. Further compound geometry testing indicated the possibility that a failure in processing any single geometric entity would lead to a total failure to process the compound geometry: when tests were repeated using individual geometry test objects all platforms reported the same results as before, with the exception of ArchiCAD 16, in which the “i”-body was accurately displayed, thus passing the “i”-body geometry tests. (This issue did not occur in the 2017 test with ArchiCAD 20, where the geometries that successfully displayed had all been correctly parsed from the single test file as with the other platforms.)

Analysis of results

Whilst in 2012 none of the tools processed all six of the geometries correctly, and the results of the tests indicated a general inherent misinterpretation of modelling of Sweeps and Boolean operations, the situation has improved in the latest (2017) tests with two of the tools correctly interpreting all the geometries. In the 2012 tests all tools consistently failed to render the “M” in BIM. Further inspection showed clear diversity in their interpretation of the buildingSMART standard. The most common problem was lack of adherence to interpreting the normal of the reference plane defined in *IfcSurfaceCurveSweptAreaSolid*. This resulted in the “M” regularly being mirrored or rotated. Several tools also failed to relocate the profile so that its origin was aligned with the first point of the directrix and its “Z” direction was tangential to that of the directrix. Whilst the standard documentation is reasonably clear it appears to have been misinterpreted. In the tests on the 2017 versions of the tools, four presented a correct rendering of the “M” in BIM. The remaining geometries all showed some form of support in each of the tools but there was little consistency. The Coordination View 2.0 specification requires that all solid geometries are transmitted as some form of *IfcSolidModel*. Boolean operations (*IfcBooleanResult*) do not fall into this scope. In order to transfer these operations (such as cutting the hole in the “B”) the standard requires an *IfcCSGSolid* to be used rather than an *IfcBooleanResult*. It was noted that more tools would support the “B” when the *IfcBooleanResult* was incorrectly used rather than when *IfcCSGSolid* was correctly used. As the geometry operation is the same this is clearly an indication that there were minor (and therefore, simple to remedy) processing errors in most of the BIM IFC readers. There was a general problem with the use of compound geometries. An *IfcShapeRepresentation* can only have one representation type; these are defined by the Standard as the types of *IfcSolidModel* shown in Table 3.

Table 3: IfcSolidModel types

Type	**Description**
Solidmodel	Including swept solid, Boolean results and *Brep* bodies more specific types are:
Sweptsolid	Swept area solids, by extrusion and revolution
Brep	Faceted *Brep* with and without voids
Csg	Boolean results of operations between solid models, half spaces and Boolean results
Clipping	Boolean differences between swept area solids, half spaces and Boolean results
Advancedsweptsolid	Swept area solids created by sweeping a profile along a directrix

There is some ambiguity as to whether *IfcSolidModel* can be used as a declaration for mixed types. This approach was adopted for the purposes of the tests, but it should be noted that some tools interpret this differently. For example, Solibri would not show an object of type *Brep* unless it was explicitly declared as a *Brep*. This causes problems when authoring complex components where the entity may be modelled as a set of different geometries; a practice that is very normal for BIM authors. One alternative - to provide different representations for each type of geometry - is an impractical one to impose on authors. The other alternative, providing a compound representation and a *Brep* representation for visual integrity, works for the viewing-only scenario but does not improve support for data reuse. In the 2012 tests all but one of the tools failed to report any errors to the user when processing the IFC file, even in the case of total failure. This is of particular concern for reasons of model validation, as those tools which aim to support model checking should consider failure to read geometry as a valid accuracy test. When the tests were repeated in 2017, with the same software platforms and with Ifc2x4, there was clear evidence that some of these problems had been overcome: all but two of the nine software tools that were re-tested in 2017 (two of the tools had been discontinued, as shown in Table 2) showed improved processing of the geometries, with two attaining a 100% result.

CONCLUSIONS

Between 2012, when the tests were first conducted, and 2017 the IFC standard itself had evolved. In 2012, the then current IFC2x3 specification had support for various kinds of solid geometry, and these definitions have been improved and enhanced in the IFC2x4 standard. Any software implementation of the standard is according to a conformance class, called a "Model View Definition" (MVD) that restricts support to a defined subset. The IFC implementations in software tools adhered to the MVD subset of the IFC2x3 specification, called "IFC2x3 Coordination View 2.0". Not all of the solid geometry types were included. For example, the *IfcSurfaceCurveSweptAreaSolid*, the CSG primitives and the *IfcCSGSolid* root were not part of the IFC2x3 Coordination View 2.0 and therefore not readily supported by most applications. As we move from the coordination view to supporting library views these geometries have become important and need to be considered.

The inconsistent support for these relatively non-complex solid geometry representations, as demonstrated by the 2012 tests, represents a problem that was likely to be compounded by an increase in the prevalence of these geometries over time. This indicated the urgency in the need to address such issues within the scope of the then current Ifc2x3 standards. The Ifc2x3 standard defined “types” of shape representation and enforced all *IfcShapeRepresentation* instances to declare their representation type. It was unclear from the standard how to interpret this representation type. Some BIM tools enforced compliance with the type rigorously: thus if the type was specified as a *Brep* but the actual object was a swept solid, the object geometry was ignored, and vice versa. However, the standard implies that if the type is specified as a *SolidModel* then all subtypes shown in Table 4 are valid. Some BIM tools obeyed this rule; others did not. It was therefore recommended that all BIM tools observe a representation type of *SolidModel* to support geometry objects that are either *SweptSolid, CSG, Brep, Clipping* or *AdvancedSweptSolid*. Where they cannot process the specified geometry a bounding box representation should be substituted with an appropriate error message. The use of *IfcCsgSolid* was found not to be correctly supported by most BIM platforms; this was the only way to create a Boolean result as a valid single solid geometry. It was therefore recommended that either *IfcBooleanResult* be a permitted *SolidModel* type within the then current Ifc2x3 standard or that implementers’ agreements require support for *IfcCSGSolid*. *IfcRevolvedSolids* appeared to be variably implemented and it was unclear what geometric constraints existed to creating these solids. The Ifc2x3 standard states “the axis shall not intersect the interior of the swept area”; this left a grey area relating to an axis which intersects the boundary of the swept area - a construction often used to create simple solids such as spheres which contain no holes. It was recommended that this be clarified in the documentation and that coincidental axis and boundary edge permitted, as this was the only way to create a sphere without the use of the unsupported CSG primitives.

Although *IfcExtrudedSolid* was shown to be generally well supported there are some aspects of the IFC documentation which could be improved to avoid problems with profiles of type *IfcArbitraryClosedProfileDef* and *IfcArbitraryClosedProfileDefWithVoid* - a geometry used to define a 2D profile that has an outer wire bound and zero or more inner loops or holes. Convention in standards such as OpenGL defines that the winding of the outer bound should be counter clockwise and the inner bounds clockwise. Whilst this is not strictly necessary to define geometric objects which have no topology (such as *IfcArbitraryClosedProfileDef*) it was considered to be useful for BIM tools to obey this convention and avoid corrections during processing. Also implicit in the documentation is that the normal of the profile shape that is to be extruded points away from the extrude direction. If this were not the case the result would be a negative solid. It was concluded that this could have been made more explicit in the technical documentation.

In summary, the consistency of support for appeared to have been poor across all BIM platforms. Relatively simple IFC geometries have been tested (there are, in fact, over 90 geometric classes that could be involved in the full execution of a compared format) and none of the BIM tools tested in 2012 gave complete support. The most likely reasons for failure were connected with incorrect interpretation of the Standard. There was demonstrable evidence that all authoring tools could natively support the tested geometries, visible through the user interface. It was thus considered that this should be a matter of software programme correction by the BIM tool author to significantly improve the level of support. The 2012 tests had revealed that model checking tools that ignored any geometry that is not correctly processed left the possibility of serious errors if they are relied upon to test actual compliance within designs for construction projects. For example, the failure to show a pipe in a clash detection check could have serious ramifications. It was therefore concluded that Bounding Box proxies should be substituted where failure-to-process occurred. Greater clarity was necessary as to which geometric representation contexts were chosen to depict an object and its geometry type and these needed to be consistent or explicit across BIM tools. In view of the critical nature of these issues, the test model used in this investigation was circulated to experts in the area, including members of BuildingSMART, and have, in part, resulted in the improved modifications to their products that have been found to be present in the 2017 tests. The tests in 2017 have shown improvements in a number of areas, with two of the BIM tools attaining 100% results. However, the fact remains that confidence in BIM modelling tools will be undermined unless interoperability improves, especially if tangible objects disappear from a project merely because a particular BIM tool has failed to process them. It is therefore further recommended that all BIM tools should indicate failure to correctly process a geometry object either in part or in full. The expectations for BIM library components within the construction industry must be supported by the functionality of existing BIM tools. These tools embody proprietary parametric components that greatly reduce geometric authoring time; however, these components have to be specifically authored for each BIM platform. For a construction product manufacturer wishing to deliver BIM library components for their product range this greatly increases the cost of authoring and maintaining those components.

ACKNOWLEDGEMENT

The authors would like to acknowledge the contribution of their former colleagues, Professor Stephen Lockley (Emeritus Professor of Building Modelling at Northumbria University, UK) and Dr. Jane Matthews (of Curtin University, Perth, Australia) in co-designing the test model and the contributing to the first software tests.

REFERENCES

BIM Task Group, British Standards Institution, Department of Business, Energy & Industrial Strategy, 2016. BIM Level 2 Standards [WWW Document]. Retrieved from: http://bim-level2.org/en/standards/

buildingSMART, 2017. Summary of IFC Releases [WWW Document]. buildingSMART-Tech.org. Retrieved from: http://www.buildingsmart-tech.org/specifications/ifc-releases/summary

buildingSMART, 2013a. IfcExtrudedAreaSolid specification [WWW Document]. Retrieved from: http://www.buildingsmarttech.org/ifc/IFC4/final/html/schema/ifcgeometricmodelresource/lexical/ifcextrudedareasolid.htm

buildingSMART, 2013b. IfcTriangulatedFaceSet specification [WWW Document]. Retrieved from: http://www.buildingsmarttech.org/ifc/IFC4/final/html/schema/ifcgeometricmodelresource/lexical/ifctriangulatedfaceset.htm

Cabinet Office, 2011. Government Construction Strategy.

Cerovsek, T., 2011. A review and outlook for a “Building Information Model” (BIM): A multi-standpoint framework for technological development. Advanced Engineering Informatics, Information mining and retrieval in design 25, 224–244.

Eastman, C.M., Eastman, C., Teicholz, P., Sacks, R., 2011. BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors. John Wiley & Sons, Hoboken, NJ.

Fischer, M., Kam, C., 2002. PM4D final report, CIFE technical report. CIFE.

Howard, R., Björk, B.-C., 2008. Building information modelling – Experts’ views on standardisation and industry deployment. Advanced Engineering Informatics, Network methods in engineering 22, 271–280.

NBS, 2017. National BIM Report 2017.

Sacks, R., Kaner, I., Eastman, C.M., Jeong, Y.-S., 2010. The Rosewood experiment — Building information modeling and interoperability for architectural precast facades. Automation in Construction, Building information modeling and interoperability 19, 419–432

Tolman, F.P., 1999. Product modeling standards for the building and construction industry: Past, present and future. Automation in Construction 8, 227–235

Treasury and Cabinet Office, 2016. Government Construction Strategy.

xBimTeam, 2017. XbimXplorer Download [WWW Document]. Xbim Toolkit Documentation. Retrieved from: http://docs.xbim.net/downloads/xbimxplorer.html

Yang, Q.Z., Zhang, Y., 2006. Semantic interoperability in building design: Methods and tools. Computer-Aided Design 38, 1099–1112.

APPENDIX 1: REPRESENTATION OF THE TESTING MODEL IN THE SELECTED APPLICATIONS

The following figures are Screenshots of the representations of the test model as computed by each of the BIM software platforms tested.

Figure 4: AECOsim from Bentley (tested 2012)

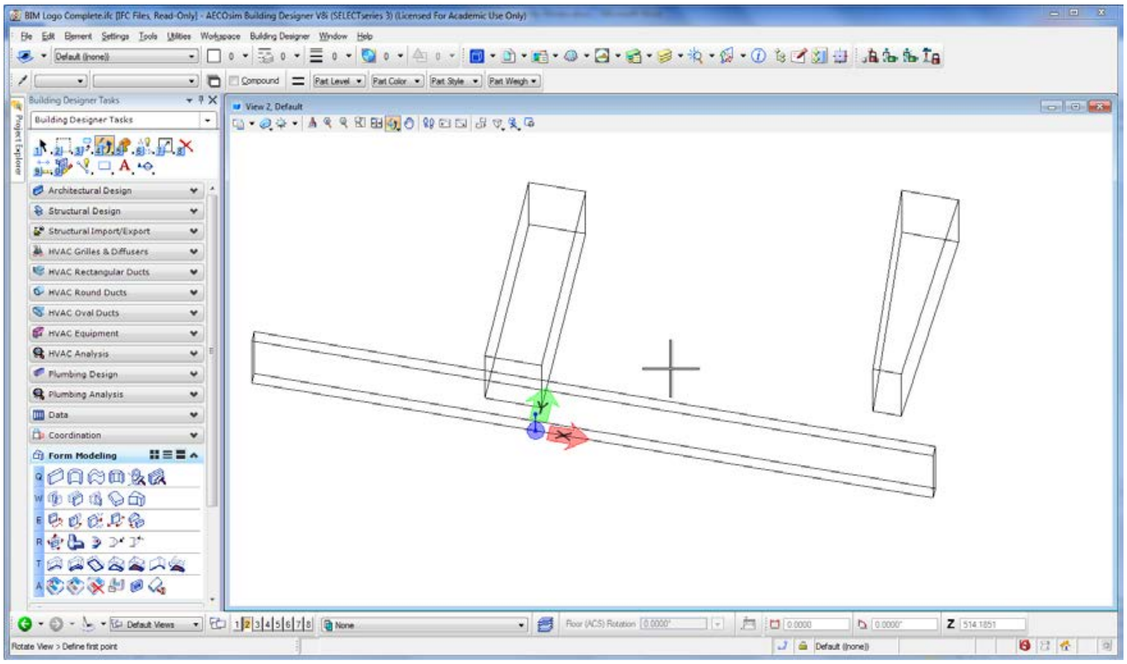


Figure 5: View v. 8i The current viewer from Bentley has been evaluated in 2017 as AECOsim was not available for testing

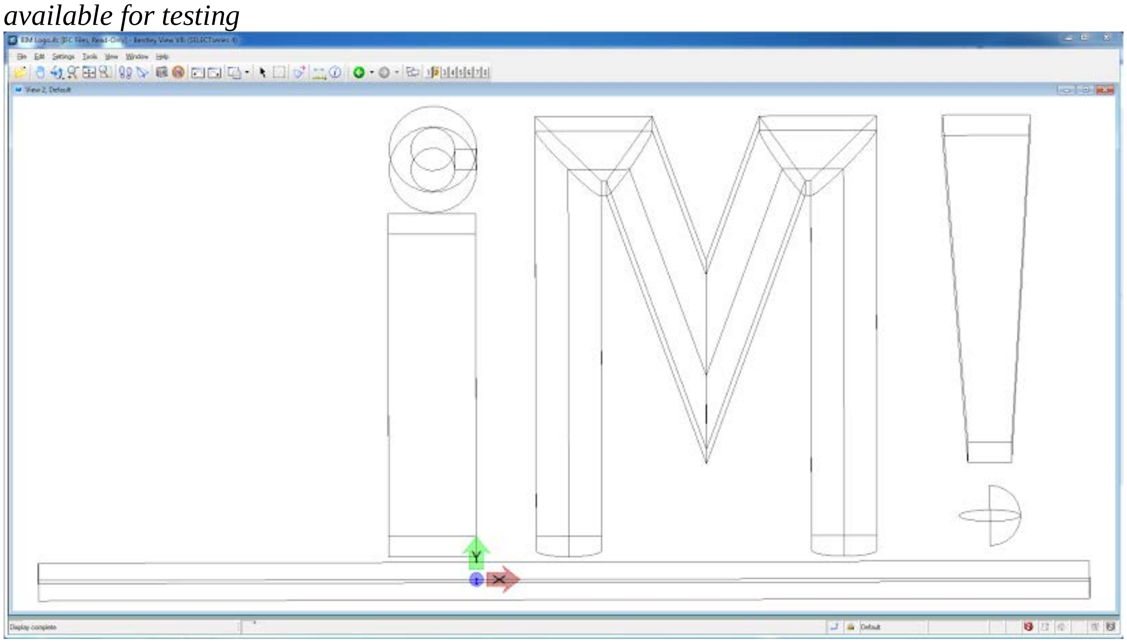


Figure 6: ArchiCAD version 16, the failure of one part affected the whole import, but more geometries were visible when primitives were loaded individually (tested 2012)

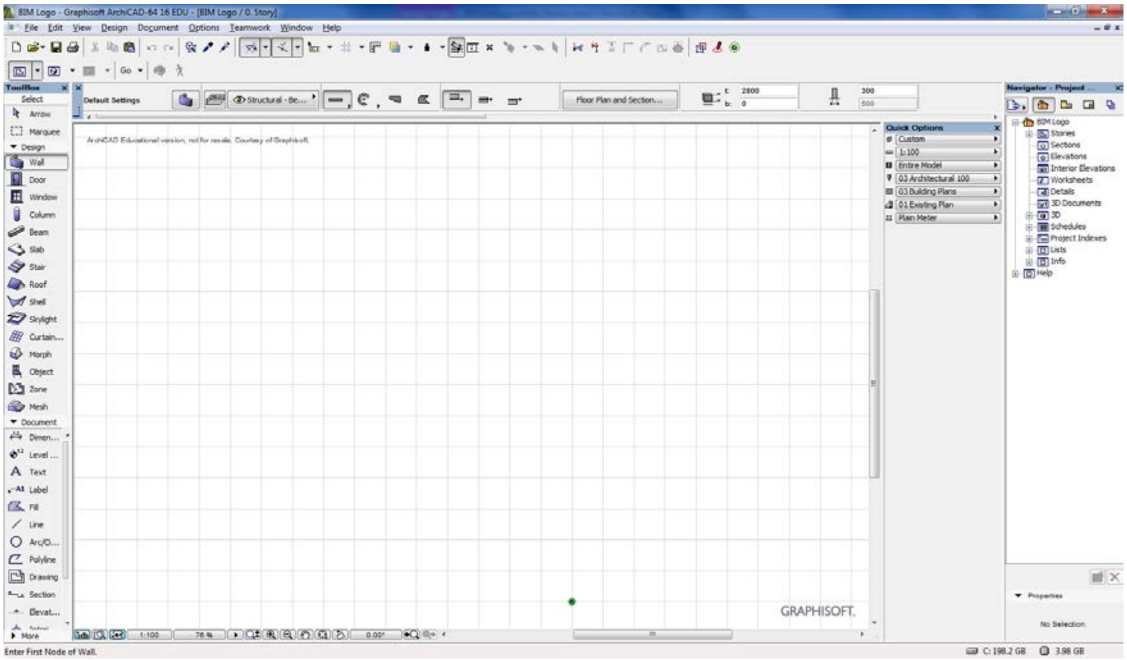


Figure 7: ArchiCAD v. 20 (tested 2017)

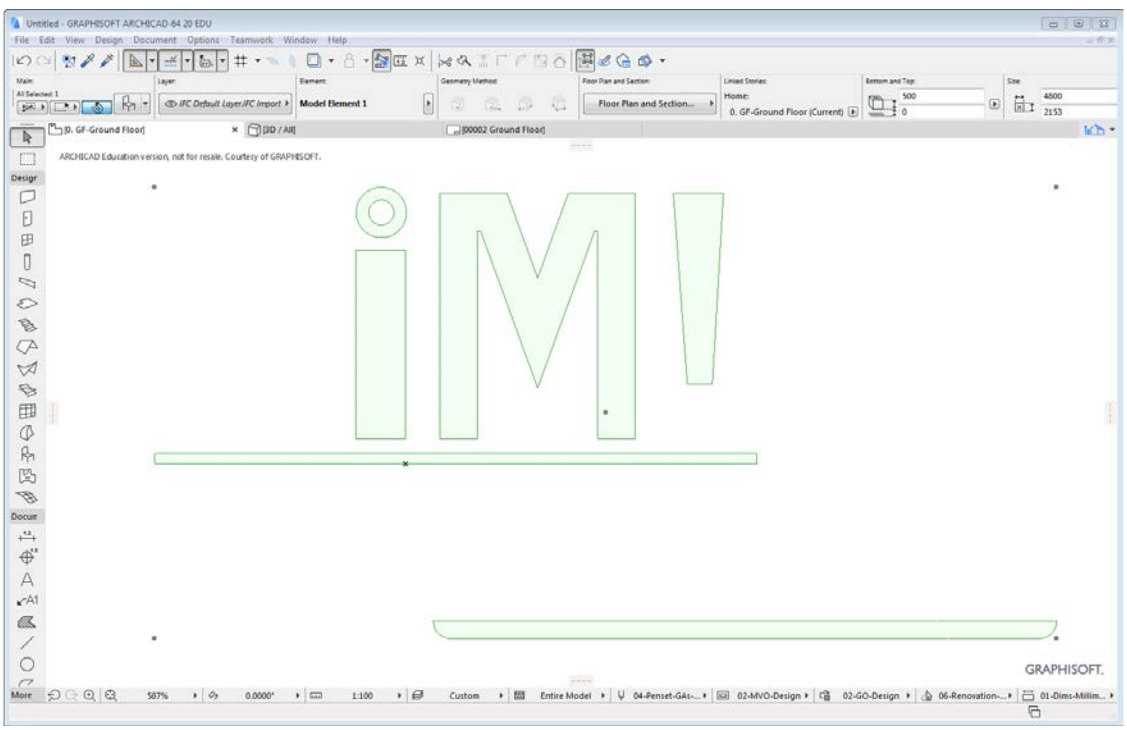


Figure 8: DDS CAD 2012 (tested 2012)

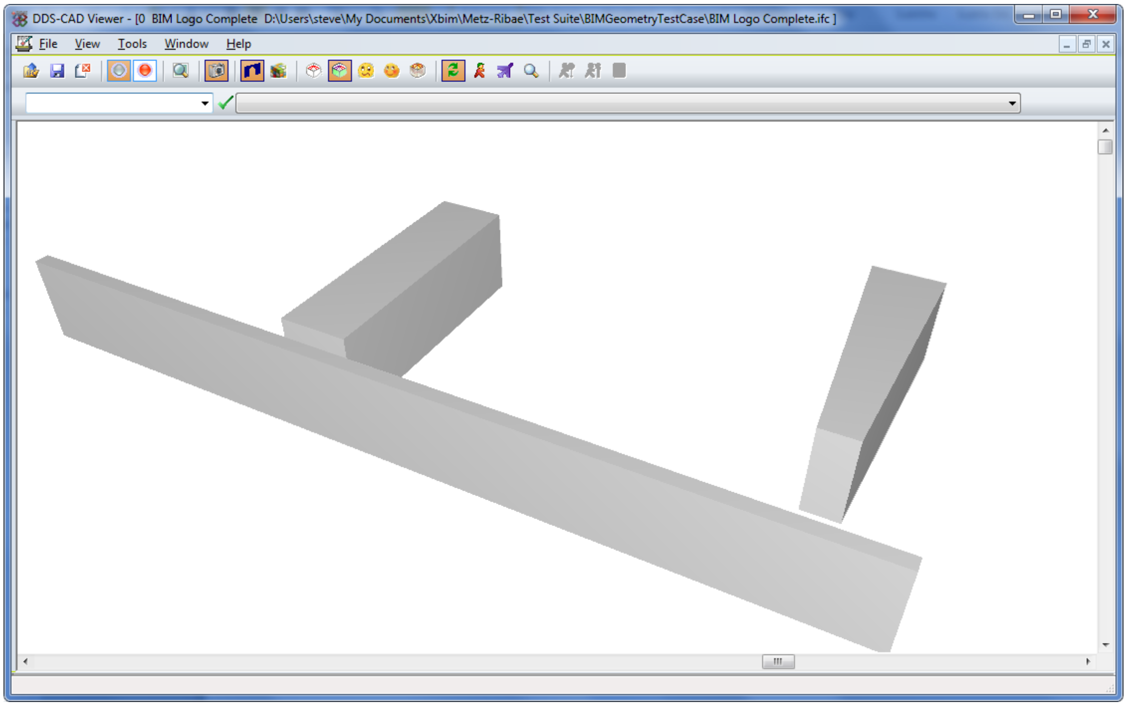


Figure 9: DDS CAD 2017(tested 2017)

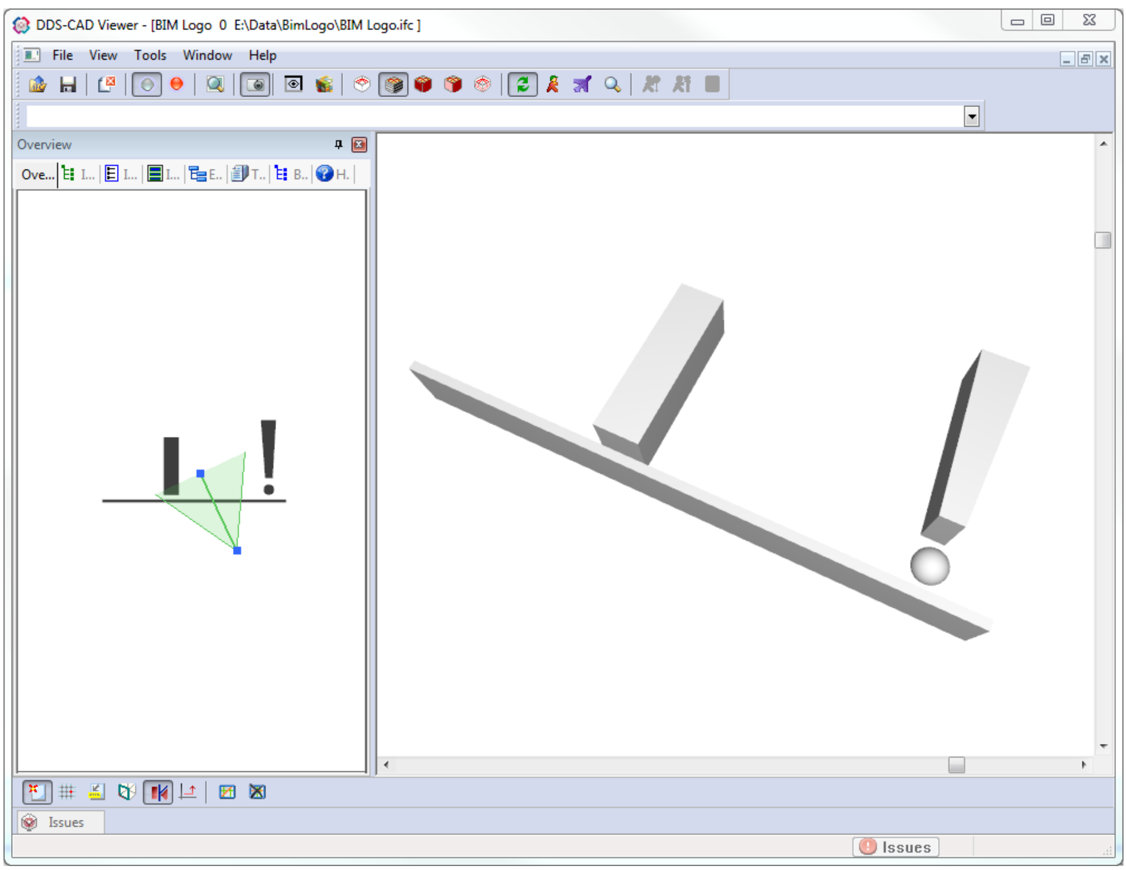


Figure 10: FZKViewer V2.2 (tested 2012)

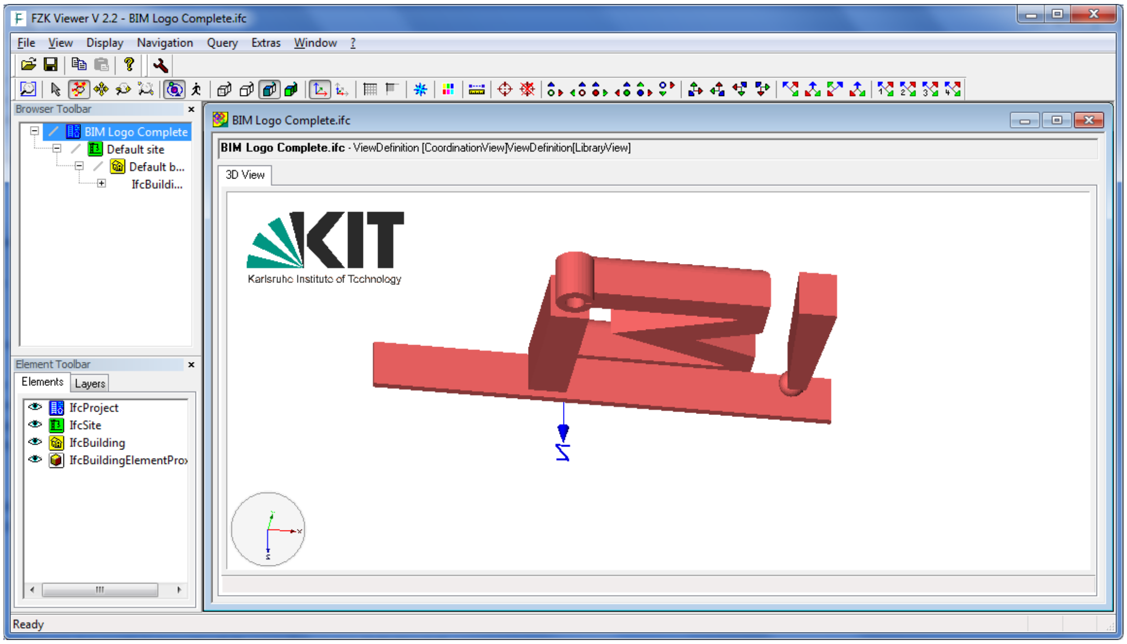


Figure 11: FZKViewer V4.8 (tested 2017)

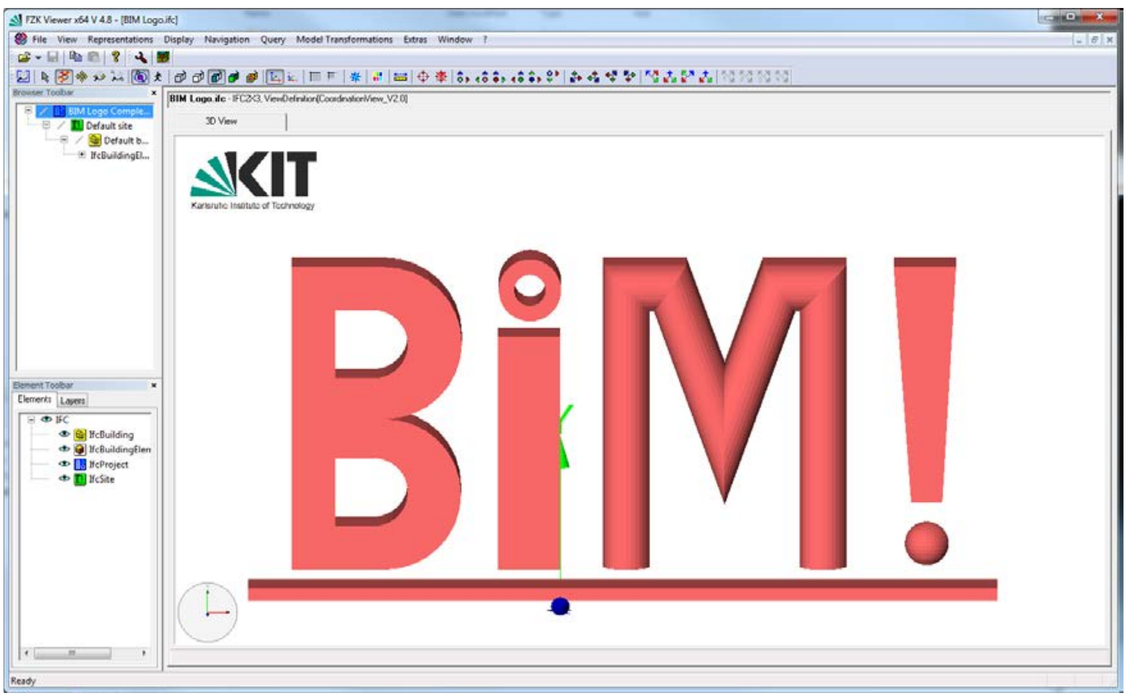


Figure 12: GTDS Viewer 0.6 (tested 2012: product discontinued before 2017)

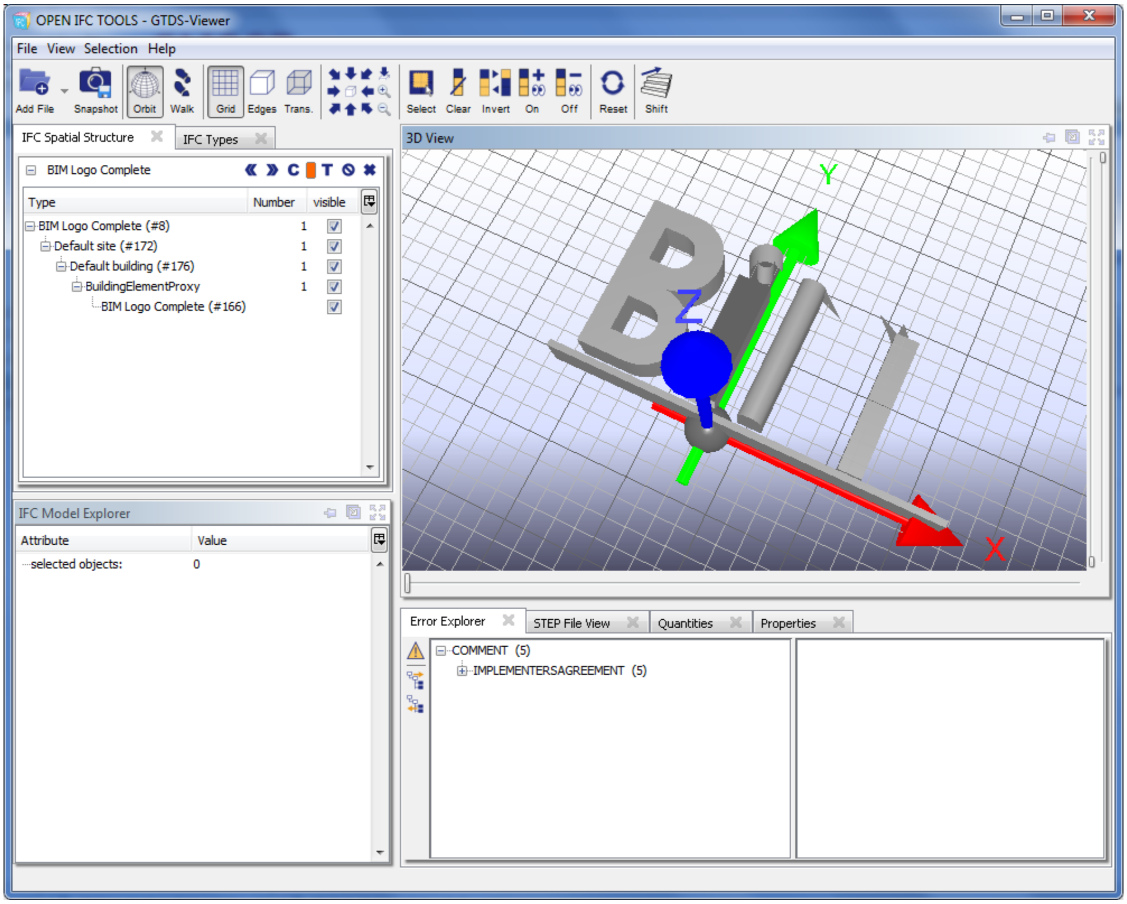


Figure 13: IfcStoreyViewer (tested 2012: product discontinued before 2017)

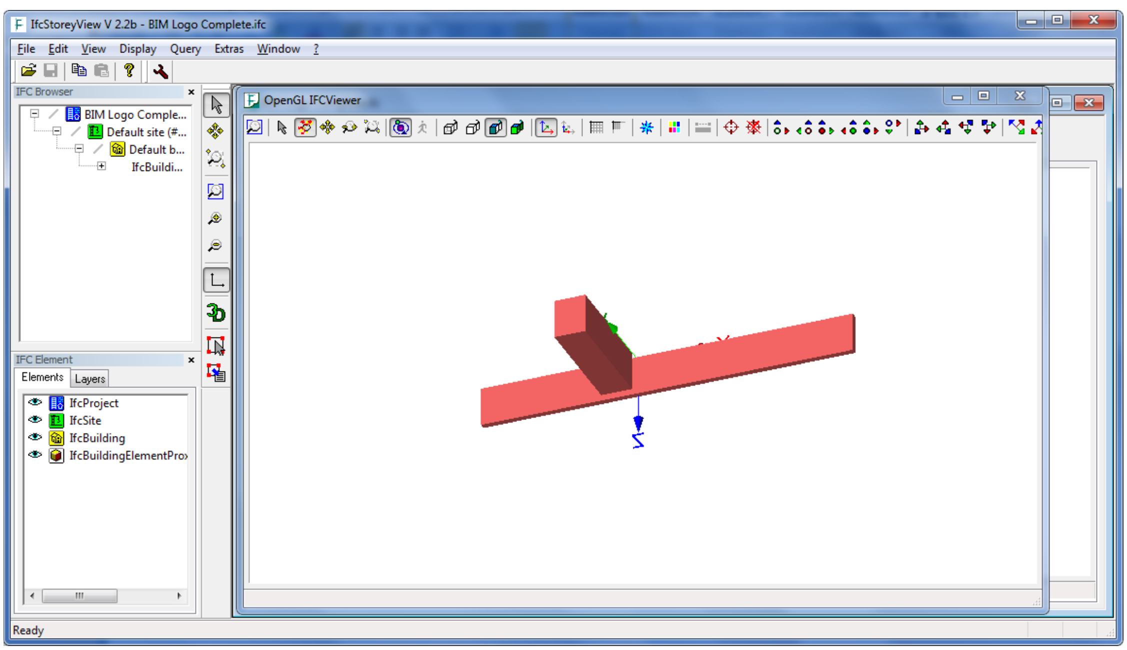


Figure 14: Revit 2012 (tested 2012)

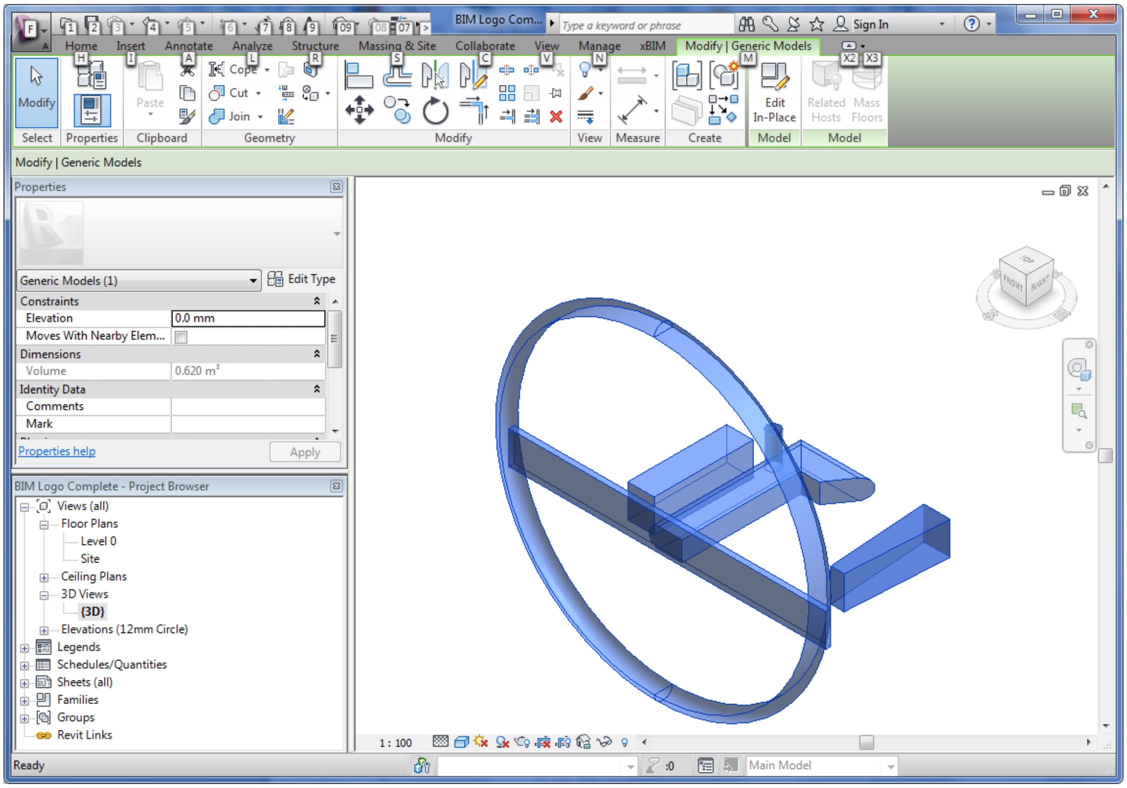


Figure 15: Revit 2017(tested 2017)

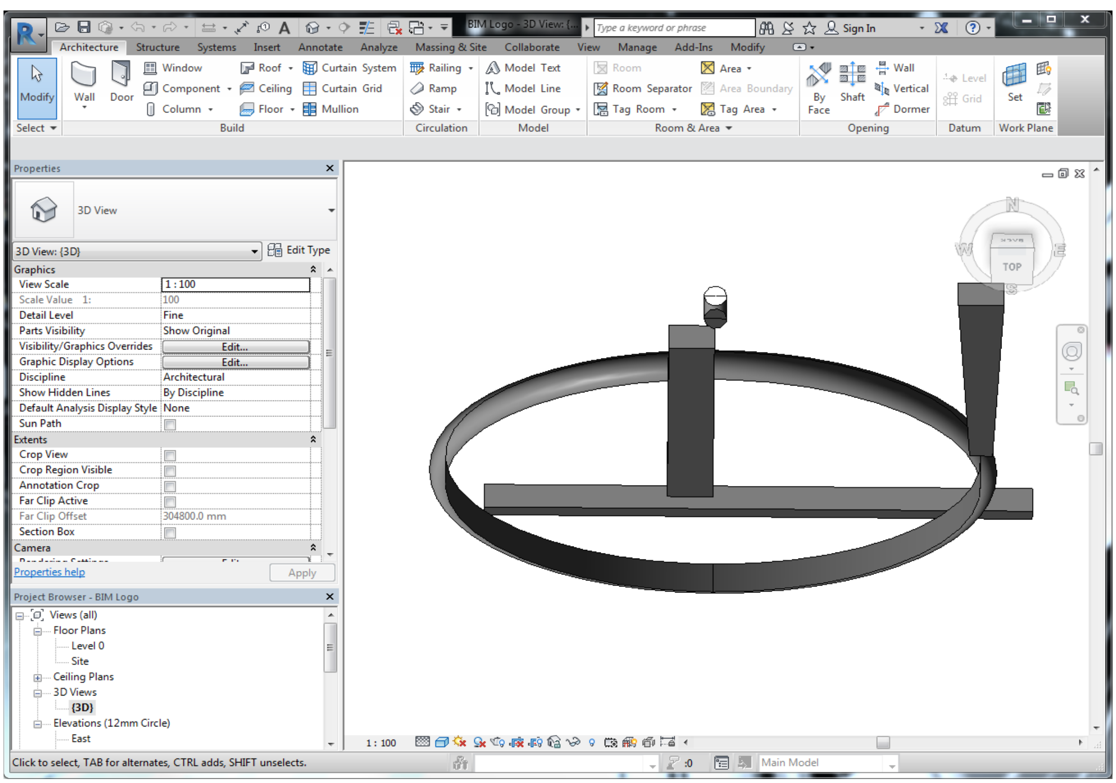


Figure 16: Solibri Model Viewer (tested 2012)

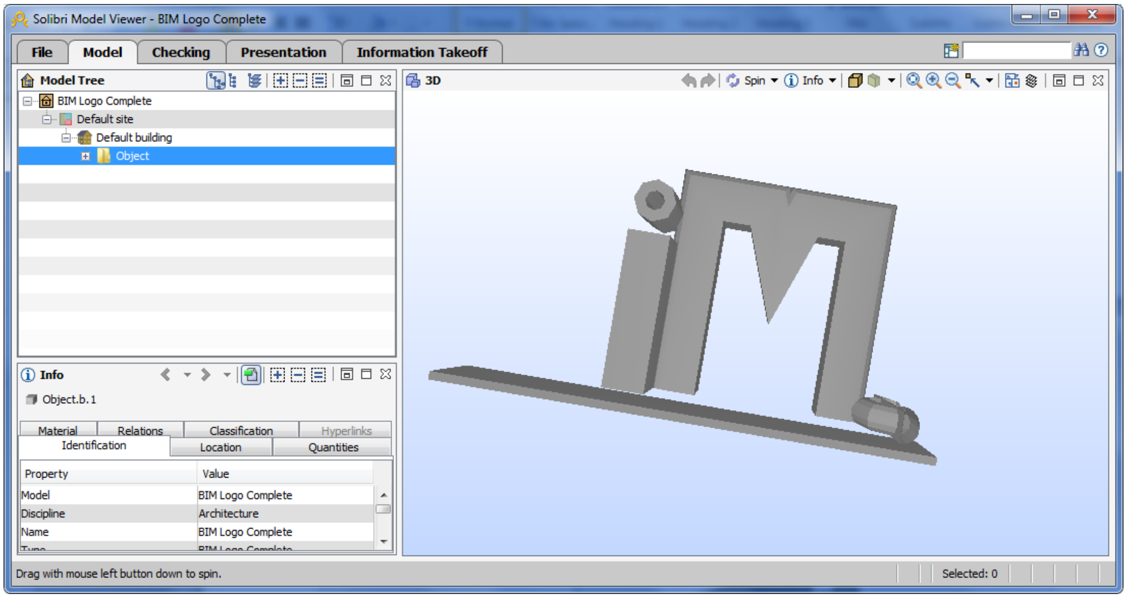


Figure 17: Solibri Model Viewer v. 9.7 (tested 2017)

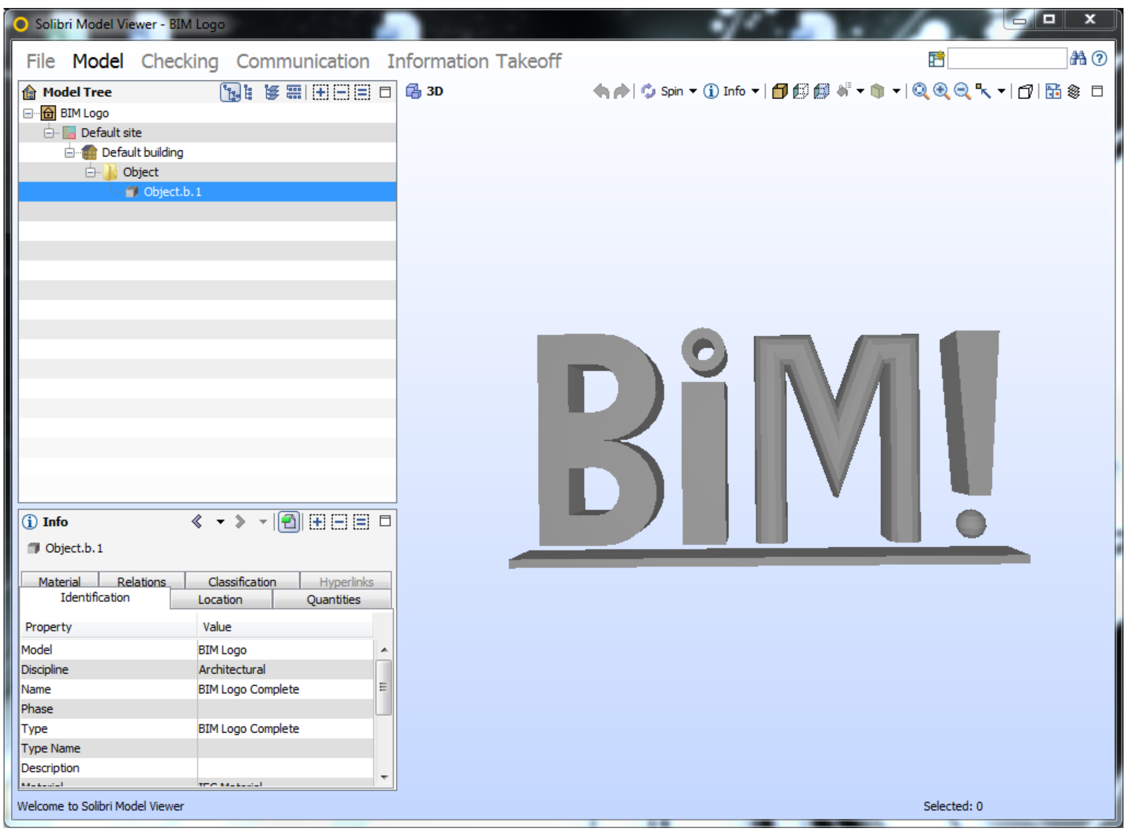


Figure 18: Tekla Structures v.18.1(tested 2012)

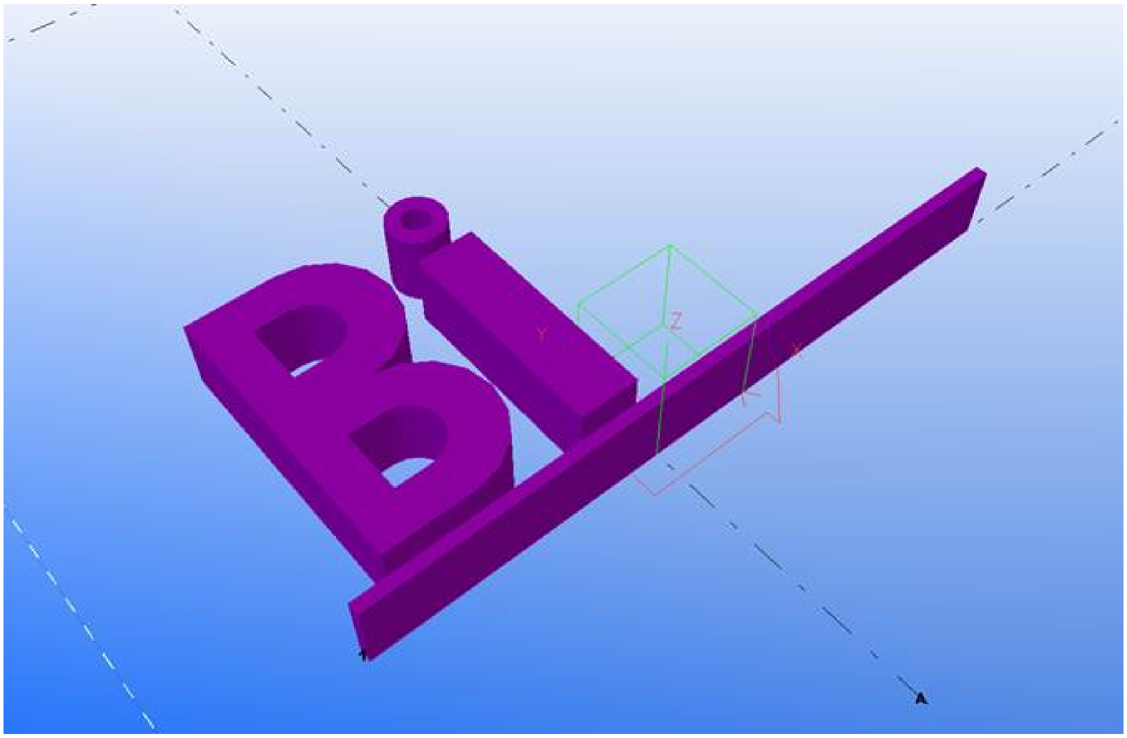


Figure 19: Tekla Structures v.2017sp2 (tested 2017)

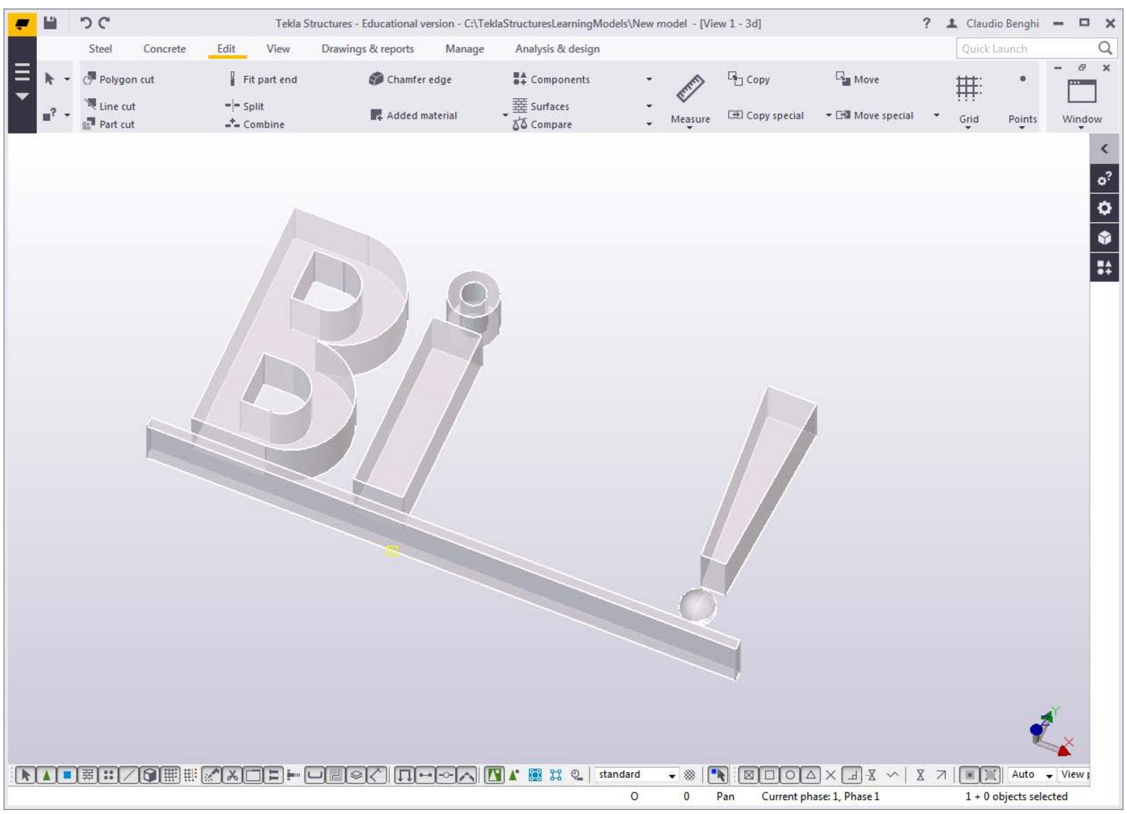


Figure 20: Tekla BIMsight (tested 2012)

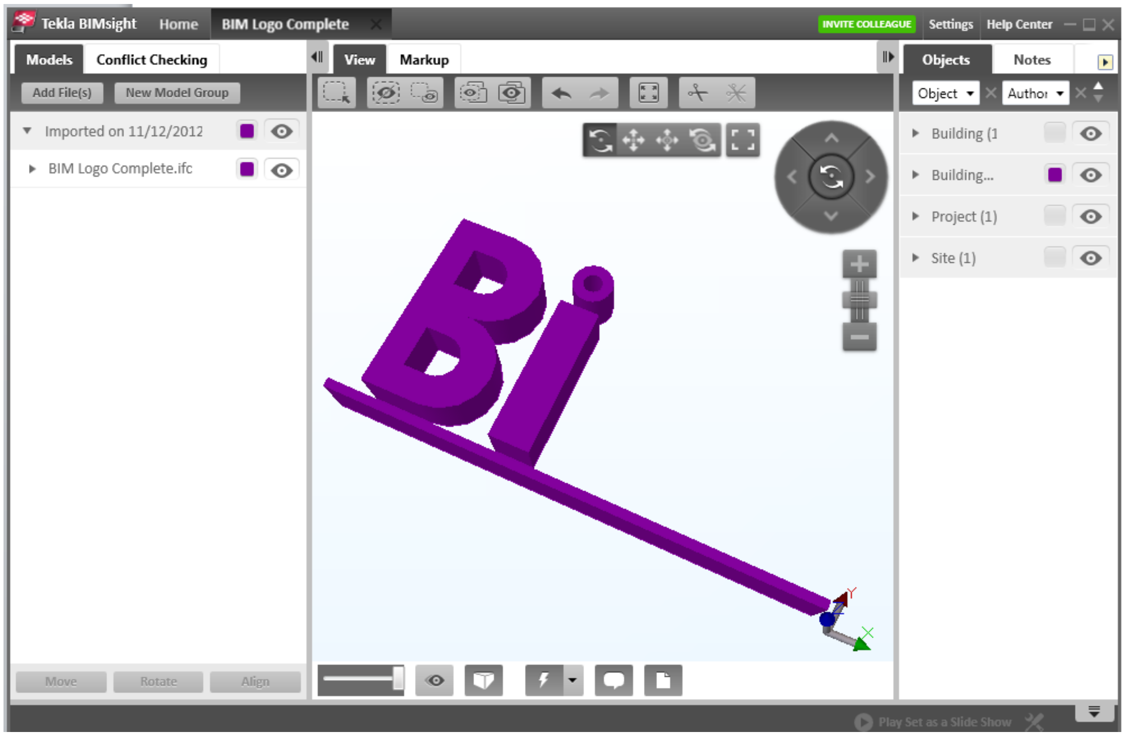


Figure 21: Tekla BIMsight (tested 2017)

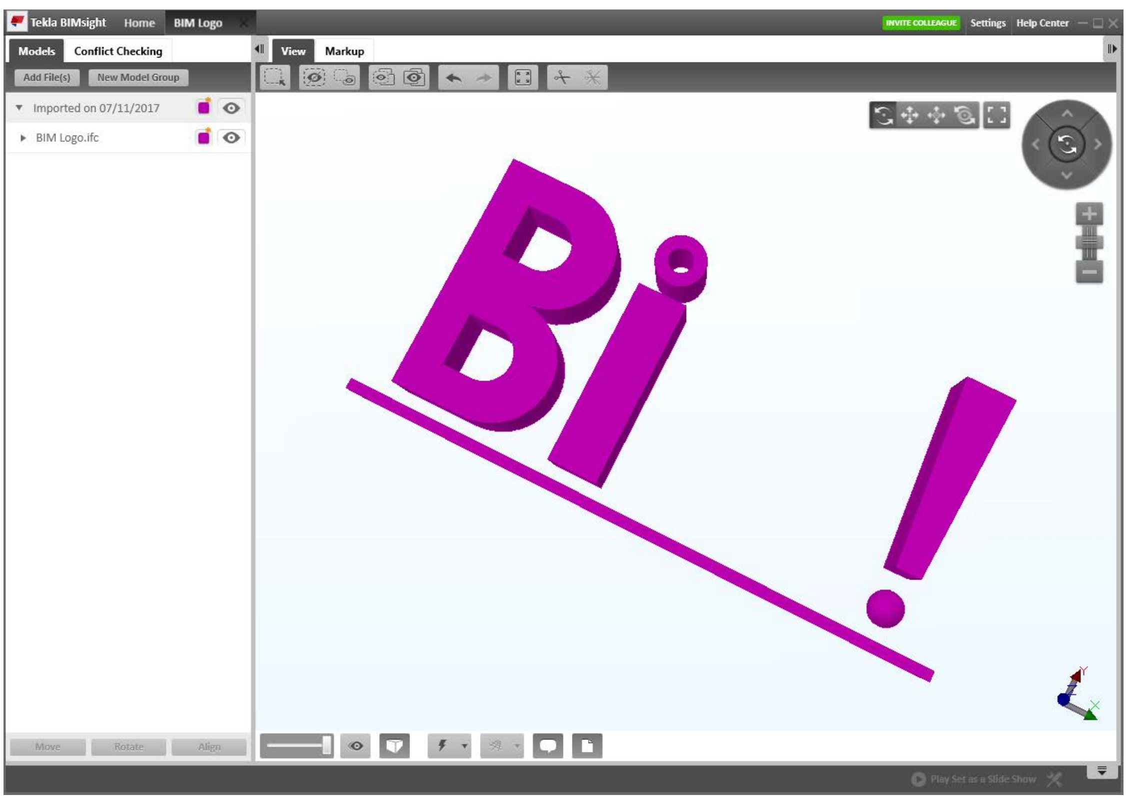


Figure 22: VectorWorks (tested 2012)

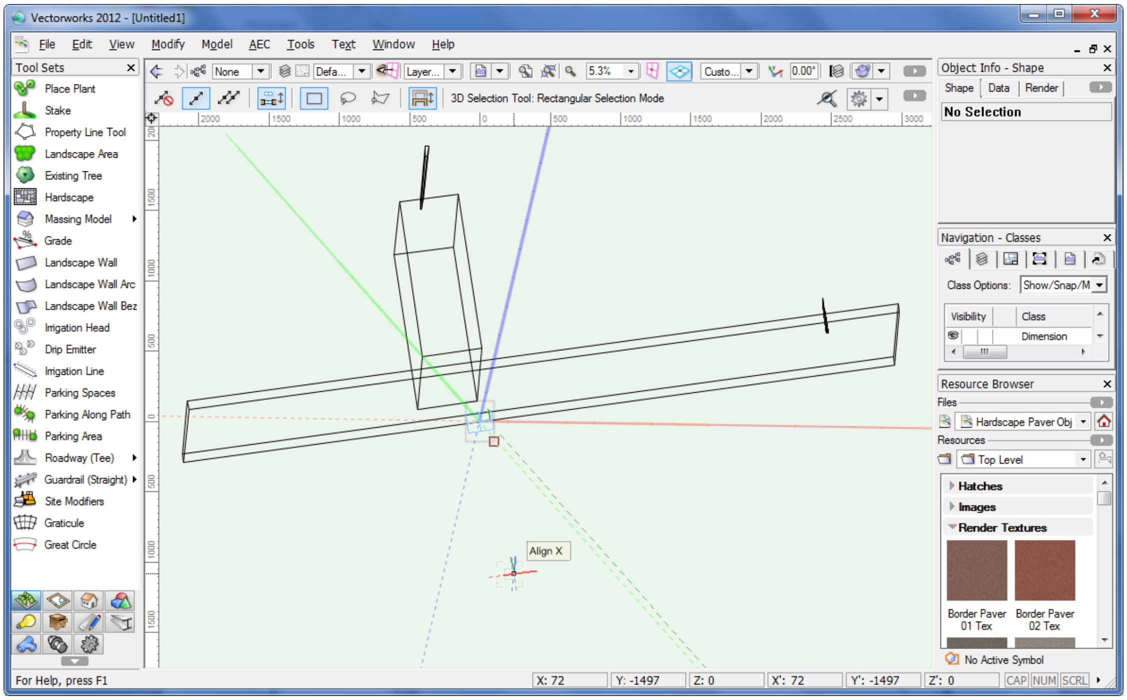


Figure 23: VectorWorks (tested 2017)

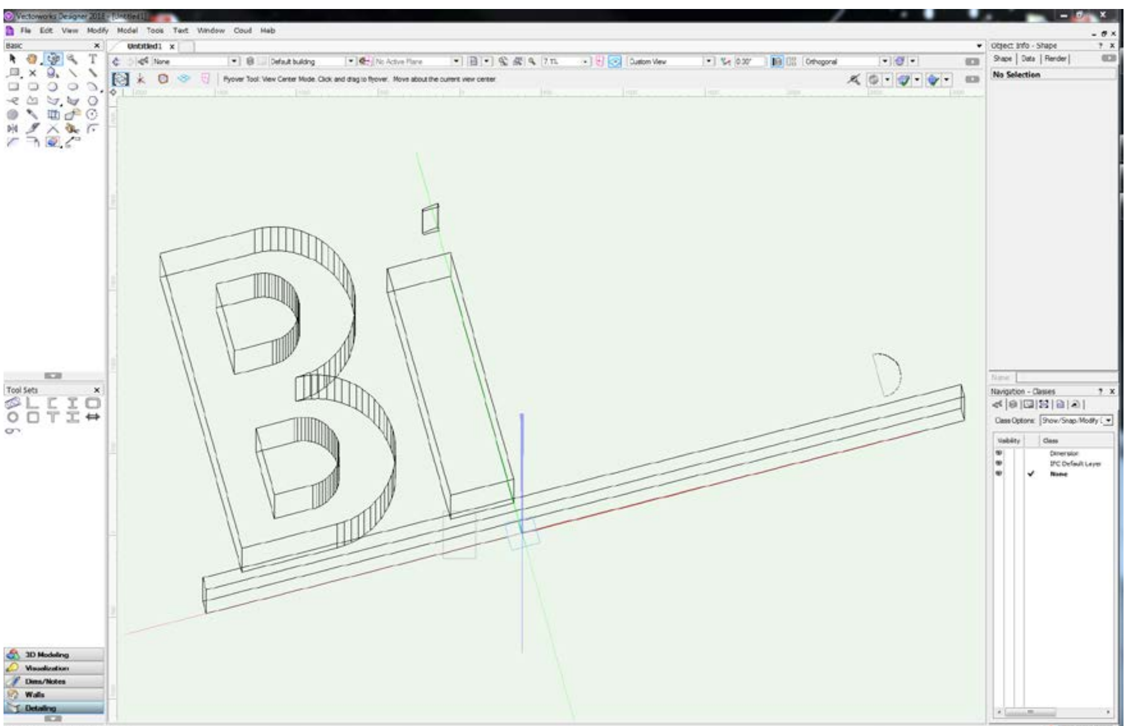


Figure 24: XbimXplorer, the items listed in the properties page display the geometric primitives used